Available online at www.sciencedirect.com

ScienceDirect

Computer Speech & Language 58 (2019) 216–230

www.elsevier.com/locate/csl

Neural sentence fusion for diversity driven abstractive multi-document summarization

Tanvir Ahmed Fuad*[,1], Mir Tafseer Nayeem[1], Asif Mahmud, Yllias Chali

University of Lethbridge, 4401 University Dr W, Lethbridge, Alberta T1K3M4, Canada

Received 16 November 2018; received in revised form 25 April 2019; accepted 28 April 2019
Available online 3 May 2019

Abstract

The lack of multi-document based models and the inaccuracy in representing multiple long documents into a fixed size vector inspired us to solve abstractive multi-document summarization. Also, there is lack of good multi-document based human-authored datasets to train any encoder-decoder models. To overcome this, we have designed complementary models for two different tasks such as sentence clustering and neural sentence fusion. In this work, we minimize the risk of producing incorrect fact by encoding a related set of sentences as an input to the encoder. We have applied our complementary models to implement a full abstractive multi-document summarization system which simultaneously considers importance, coverage, and diversity under a desired length limit. We conduct extensive experiments for all the proposed models which bring significant improvements over the state-of-the-art methods across different evaluation metrics.

2000 MSC: 68T50

Keywords: Abstractive multi-document Summarization; Sentence fusion; Neural fusion model; Document clustering

1. Introduction

The automatic document summarization systems aim at finding the most relevant information in a text and presenting them in a condensed form. There are two types of summarizations: abstractive summarization and extractive summarization. Abstractive summarization systems define the actual meaning of the given documents. For this purpose, it needs to understand the whole document and then create the summary accordingly. To do so, it needs extensive natural language generation, for which it has to use paraphrasing, generate words and restructure the sentences which made it highly complex. Summarization systems are classified as single-document or multi-document based on the number of source documents. The information overlap between the documents of the same topic makes the multi-document summarization more challenging than the task of summarizing single document.

Recent success of neural sequence-to-sequence (**seq2seq**) models provide an effective way for text generation

* Corresponding author.
E-mail address:
[1] Fuad and Nayeem have equal contributions.

https://doi.org/10.1016/j.csl.2019.04.006
0885-2308/

which achieved remarkable success in the case of abstractive sentence summarization (Rush et al., 2015; Nallapati et al., 2016; Zhou et al., 2017; Suzuki and Nagata, 2017) using English Gigaword dataset (Napoles et al., 2012). The output generated by these models are very short (about 75 characters), ungrammatical and sometimes produce fake facts (Cao et al., 2018). Moreover, neural abstractive summarization models have outperformed extractive and abstractive methods (See et al., 2017; Narayan et al., 2018a; 2018b; Fan et al., 2017; Celikyilmaz et al., 2018) on single document summarization task with huge training data from CNN/DailyMail corpus. Very recently, the experimental results of (Celikyilmaz et al., 2018) reveal that the current neural single document summarization models suffer from common mistakes such as missing key facts, reporting inaccurate fact, repeating the same content, and including unnecessary details. The encoding of a collection of related documents and even single document still lack satisfactory solutions due to the long range dependencies of RNNs. Unfortunately, the extension of **seq2seq** models to MDS (Multi-Document Summarization) is not straightforward due to the lack of large multi-document summarization datasets needed to train the computationally expensive sequence-to-sequence models. In this paper, we tackle these aforementioned issues by encoding a related set of sentences as input to the encoder to minimize the risk of producing incorrect facts. In the process of developing model for MDS, we have also developed new models for two important NLP tasks such as sentence clustering and neural sentence fusion. For these tasks we have used a bi-directional GRU for the clustering and transformer for the sentence fusion. We have built the clustering model to maintain the diversity among the texts and also ensure the information diversity. Using transformer for the clustering task could give us slightly better result, but it is a rich model and it is computationally expensive. Here, our main concern is the MDS task, so we have not considered transformer for the clustering model to ensure both time and efficiency.

2. Related works

Recently, end-to-end training with encoder-decoder neural networks have achieved notable success in case of abstractive summarization. These systems have adopted techniques such as encoder-decoder with attention (Bahdanau et al., 2015; Luong et al., 2015) neural network models from the field of machine translation to model the sentence summarization task. Rush et al. (2015) was the first to use neural sequence-to-sequence learning in headline generation task from a single document. Unfortunately, this area of research under the term sentence summarization (Rush et al., 2015), which can generate only a single sentence, somewhat misleadingly called text summarization in some follow-up research works (Nallapati et al., 2016; Chopra et al., 2016; Suzuki and Nagata, 2017; Zhou et al., 2017; Ma et al., 2017). However, there has been some recent attempts which use CNN/DailyMail corpus (Hermann et al., 2015) as a supervised training data to generate multi-sentence summary from a single document (See et al., 2017; Li et al., 2017b; Paulus et al., 2017; Narayan et al., 2018a; 2018b; Fan et al., 2017; Celikyilmaz et al., 2018). The recent abstractive summarization models actually produce compressive summaries by deleting the words form a single source document, no direct paraphrasing is involved in the process. Hence, no new words are generated which is different form the source document words (other than morphological variations), which is pointed out by their own experimental results. Recently, some researchers employ neural network based framework to tackle the summarization problem in multi-document setting (Yasunaga et al., 2017; Li et al., 2017a). Yasunaga et al. (2017)'s method is limited to extractive summmarization. On the other hand, Li et al. (2017a)'s method is limited to compressive summary generation using an ILP based model, and there is no explicit redundancy control in the summary side. Zhang et al. (2018) proposed a neural network based solution for abstractive MDS task by adding a document set encoder for a set of documents. Encoding a large set of documents into a fixed size vector produces highly incorrect facts. As a result, their model fails to compete with several simple baselines for this task.

3. Proposed model

3.1. Sentence clustering

Text clustering is a challenging problem due to its sparseness of text representation as most words only occur once in a text (Aggarwal and Zhai, 2012). As a result, the Term Frequency-Inverse Document Frequency (TF-IDF) measure does not work well. In order to address this problem, we use word embedding and deep neural network architectures for better representation of text and hence propose an unsupervised sentence clustering model.

3.1.1. Proposed method

A sentence is a sequence of words $\mathbf{S} = (w_1, w_2, \ldots, w_L)$, where L is the length of the sentence. We encode a sentence using bi-directional GRUs (Cho et al., 2014). While training the GRU we have used binary cross-entropy, loss function and adam optimizer; also, the dropout was set to 0.01 and activation layer was tanh. The GRU (Cho et al., 2014) achieves similar performance as LSTM (Hochreiter and Schmidhuber, 1997) but it is fast, computationally efficient and can improve performance on long sequences. In the simplest uni-directional case, while reading input symbols from left to right, a GRU learns the hidden annotations h_t at time t,

$$h_t = \mathbf{GRU}\Big(h_{t-1}, e(w_t)\Big). \tag{1}$$

where, the $h_t \in \mathrm{IR}^n$ encodes all content seen so far at time t which is computed from h_{t-1} and $e(w_t)$, where $e(w_t) \in \mathrm{IR}^m$ is the m-dimensional embedding of the current word w_t. We can use any pre-trained word vectors as input to GRUs.

In our work, we apply bi-directional GRUs (bi-GRUs), which we found to give better results than single directional GRUs consistently. As shown in Figure 1, Bi-GRU processes the input sentence in both forward and backward direction with two separate hidden layers calculated with GRUs, obtains the forward hidden states ($\overrightarrow{h}_1$,..., $\overrightarrow{h}_L$) and the backward hidden states ($\overleftarrow{h}_1$,..., $\overleftarrow{h}_L$). For each position t, we simply concatenate both forward and backward states into the final hidden state:

$$h_t = \overrightarrow{h}_t \oplus \overleftarrow{h}_t. \tag{2}$$

in which operator $\oplus$ indicates concatenation. $\overrightarrow{h}_t$ is calculated using equation (1) and $\overleftarrow{h}_t$ is calculated using the following equation:

$$\overleftarrow{h}_t = \mathbf{GRU}\Big(\overleftarrow{h}_{t+1}, e(w_t)\Big). \tag{3}$$

$\overrightarrow{h}_0$ is initialized as zero vector, and the output sentence embedding x_i for the sentence S_i is the last hidden state:

$$S_i = x_i = h_L. \tag{4}$$

Inspired from Murtagh and Legendre (2014)'s method, we use a hierarchical clustering algorithm with a complete linkage criteria. This algorithm proceeds incrementally, starting with each sentence considered as a cluster, and

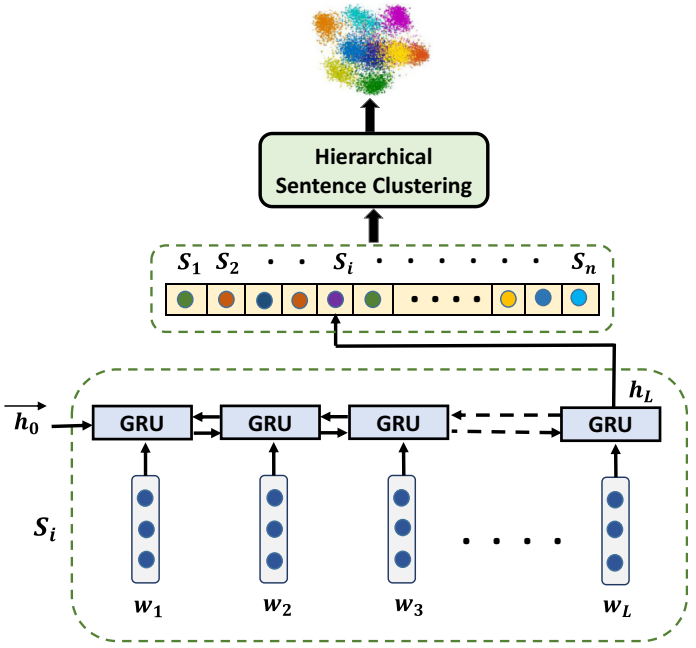

Fig. 1. Sentence clustering model.

merging the pair of similar clusters after each step using bottom up approach. The complete linkage criteria determines the metric used for the merging strategy, which means largest distance between a sentence in one cluster and a sentence in the other candidate cluster. While building the clusters, we use the cosine similarity between the sentence embeddings obtained from Eq. (4). We set a similarity threshold ($\tau = 0.5$) to stop the clustering process by using a hold out dataset SICK[1] of SemEval-2014 (Marelli et al., 2014) for getting optimal performance. If we cannot find any cluster pair with a similarity above the threshold ($\tau = 0.5$), the process stops, and the clusters are released.

3.2. Neural sentence fusion

Multi-sentence compression (**MSC**) usually takes a group of related sentences and produces an output sentence by merging the sentences about the same topic. MSC is a text-to-text generation process in which a novel sentence is produced as a result of summarizing a set of similar sentences originally called sentence fusion (Barzilay and McKeown, 2005). Recent success of neural sequence-to-sequence (**seq2seq**) models provide an effective way for text generation which achieved remarkable success in case of abstractive sentence summarization which can perform deletion based compression from a single source sentence (Rush et al., 2015; Nallapati et al., 2016; Zhou et al., 2017; Ma et al., 2017). As MSC is a text-to-text sentence generation process which creates new words and sentence structures, it has the amenable capability of improving the abstractiveness. Moreover, there are some recent attempts which use CNN/Daily Mail corpus (Hermann et al., 2015) as a supervised training data to generate multi-sentence summary from a single document (See et al., 2017; Li et al., 2017b; Paulus et al., 2017; Fan et al., 2017) using neural architectures. In this work, we investigate applying **seq2seq** encoder-decoder models to the case of **MSC** task. Our task is completely different, it takes a related ordered set of sentences and produces a single output sentence by fusing or merging the input sentences instead of encoding a single sentence or a document. The main difference between sentence fusion and encoder-decoder summarization is that, in fusion we will only get a single sentence as the output, but in encoder-decoder summarization we might get multiple sentences. To the best of our knowledge, our work is the first to investigate adapting deep neural network for sentence fusion task.

3.2.1. Proposed Method

Given a related set of source sentences about a topic $\mathbf{X} = (X_1, X_2, \ldots, X_N)$, our model learns to predict its abstractive multi-sentence compression target $Y = (y_1, y_2, \ldots, y_M)$, where $N > 1$ and $M < |X_1| + |X_2| + ... + |X_N|$. In this work, we use the **Transformer** model (Vaswani et al., 2017) which has significantly improved state-of-the-art models for a wide variety of applications, such as machine translation, parsing, image captioning and more. The **Transformer** follows the overall architecture for a standard encoder-decoder model, replacing the complex recurrent or convolutional layers most commonly used in encoder-decoder architectures with multi-headed self-attention. The natural ability of multi-head attention mechanism is to jointly attend to similar phrases from different positions of a sequence makes this an ideal choice for our model. In this method we have chosen **Transformer** over seq2seq because of it's efficiency and working process. Usually RNN or CNN handles a sentence as a sequence, meaning word by word sequentially, which sometimes induces vanishing gradient problem. On the other hand **Transformer** works with the whole sentence at once as a single package in $O(1)$ time. In our work we have used the clusters and it was important to carry the fact from our sentences. So that, we have preferred to use **Transformer** here, because it ensures the efficiency while keeping the most important facts. We have used the implementation provided by the authors[2]. We keep the exact same settings which were suggested for summarization. More details about multi-head attention and overall architecture can be found in (Vaswani et al., 2017).

3.3. Multi-document summarization

We use our sentence clustering technique proposed in Section 3.1 to group related sentences from the document set on a given topic. We then order the clusters and the sentences inside the clusters using the heuristic sentence ordering techniques presented in Section 3.3.1.1. For each cluster of related ordered sentences, we use our neural sentence fusion model presented in Section 3.2 to generate fused abstractive versions of the multiple related senten-

[1] http://clic.cimec.unitn.it/composes/sick.html
[2] https://github.com/tensorflow/tensor2tensor

ces extracted form the document set. Finally, we use our ILP based abstractive sentence selection mechanism which is presented in Section 3.3.1.2 to select the best subset of sentences which simultaneously considers importance, coverage and diversity under a desired length limit. The overall process is presented in Fig. 2.

3.3.1. Proposed method

In this section we have presented the idea of our MDS model.

3.3.1.1. Sentence ordering. A wrong order of sentences convey entirely different idea to the reader of the summary. In a single document, summary information can be presented by preserving the sentence position in the original document. In multi-document summarization, we can't directly use the sentence position as the sentences are coming form a set of documents. Therefore, we implement two cluster ordering techniques that reorder clusters based on the original sentence position in the documents.

Intra-cluster ordering: The sentences $\{S_1, S_2, \ldots, S_i, .., S_n\}$ in any cluster C_i are assigned a normalized score. For example, the normalized score of S_i is computed as the ratio of the original position of the sentence and the total number of sentences in document D_i (here, S_i belongs to document D_i). We pass this ordered related set of sentences to our neural sentence fusion model.

Inter-cluster ordering: When ordering two different clusters, the cluster that has the lower score obtained by averaging the normalized scores of all the sentences in that particular cluster is ranked higher than the others.

3.3.1.2. Abstractive sentence selection. In this work, we use the concept-based ILP framework introduced in Gillick and Favre (2009)'s publication with some suitable changes to select the best subset of sentences. We propose an ILP based sentence selection mechanism which integrates three important measures namely importance, coverage, and

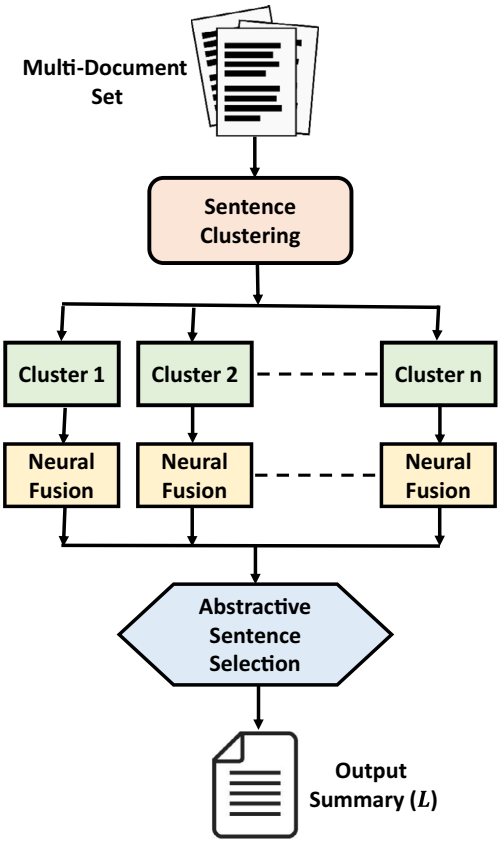

Fig. 2. Multi-document summarization model.

diversity to extract the sentences for the summary under a certain length limit.

3.3.1.3. Importance. One of the basic requirements of a good summary is that it should contain the most important information across multiple documents. To model this property, we use bi-grams. Bi-grams are the phrases that represent the main topics of a document. Sentences containing the most relevant phrases are important for the summary generation. We assign a weight to each bi-gram using its document frequency. Let w_i be the weight of bi-gram i and b_i a binary variable that indicates the presence of bi-gram i in the extracted sentences. We try to maximize the weight of the bi-grams in the selected summary sentences as follows:

$$S_{imp} = \sum_i w_i b_i. \tag{5}$$

3.3.1.4. Coverage. A good summary has the capability to cover most of the important aspects of a document set. To formulate this, we select at most one sentence from the cluster of related sentences to increase the information coverage from the document side. In order to ensure at most one sentence per cluster in the selected sentences we add an extra constraint in our overall ILP formulation like the following equation, where g_c a cluster of sentences that corresponds to the set of similar sentences.

$$\sum_{j \in g_c} s_j \leq 1, \quad \forall g_c \tag{6}$$

3.3.1.5. Diversity. Maximizing diversity in the summary is another basic requirement in any summarization task. We define the degree of diversity of a generated summary by measuring the dissimilarity among the selected sentences. Let the generated summary is Y and $|Y|$ is the total number of sentences in the summary. We compute S_{div} as the mean of the pairwise dissimilarities among the selected sentences.

$$S_{div} = \frac{1}{|Y|\ (|Y|-1)} \sum_{i \in Y} \sum_{j \in Y} d(S_i, S_j). \tag{7}$$

where $d(..)$ is the dissimilarity function calculated by

$$d\ (S_i, S_j) = 1 - \frac{S_i \cdot S_j}{\| S_i \| \quad \| S_j \|}. \tag{8}$$

Intuitively, the more diverse (or more dissimilar) the selected sentences to each other, the higher the diversity.

3.3.1.6. Summary length limit. One of the essential properties of the text summarization systems is the ability to generate a summary with a fixed length, which has a common commercial use case (e.g., 160 to 300 characters for search result and news article summarization by news aggregators, especially on mobile devices). Recently, Fan et al. (2017) presents a neural model that enables users to specify desired length in order to control the shape of the final summaries which is only limited to single document summarization. In this paper, we address this issue in multi-document setting, our model can generate summaries given a desired length.

Finally, we propose an ILP formulation which considers the above mentioned aspects in context of multi-document summarization. The final summaries are generated by assembling the optimally selected sentences. Let l_j be the number of words in sentence j, s_j a binary variable that indicates the presence of sentence j in the selected sentence set and L the length limit for the summary. Let Occ_{ij} indicate the occurrence of bi-gram i in sentence j, the final ILP formulation is,

$$Maximize : S_{imp} + S_{div}. \tag{9}$$

$$Subject\ to : \sum_j l_j s_j \leq L. \tag{10}$$

$$s_j Occ_{ij} \leq b_i, \quad \forall i, j \tag{11}$$

$$\sum_j s_j Occ_{ij} \geq b_i, \quad \forall i \tag{12}$$

$$\sum_{j \in g_c} s_j \leq 1, \quad \forall g_c \tag{13}$$

$$b_i \in \{0, 1\} \quad \forall i \tag{14}$$

$$s_j \in \{0, 1\} \quad \forall j \tag{15}$$

We try to maximize the importance score as well as the diversity in the output summary sentences (9), while avoiding repetition of those bi-grams (11, 12) and staying under the maximum number of words allowed for the summary (10). We select at most one sentence from the cluster of related sentences to increase the information coverage form the multi-document point of view. In this process, we extract the optimal combination of sentences as output summary.

4. Experiments

In this section we have represented the baselines, datasets and the results of our models.

4.1. Clustering

4.1.1. Datasets

- *StackOverflow*[3] We use the challenge data published in Kaggle.com[4]. This dataset consists of 3,370,528 samples from July 31st, 2012 to August 14, 2012. In our experiments, we randomly select 20,000 question titles from 20 different tags.
- *SearchSnippets*[5] This dataset was constructed from the different predefined phrases of web search transaction results of 8 different domains (Phan et al., 2008).

4.1.2. Pre-trained word vectors

The word embeddings are low dimensional vector representations of words such as **word2vec** (Mikolov et al., 2013) and **GloVe** (Pennington et al., 2014) which recently gained much attention in various natural language processing tasks. Recently, Bojanowski et al. (2017) proposed a simple method named **fastText** to learn word representations by taking into account sub-word information. We conduct extensive experiments for our model (**HierGRU**) on two public datasets with these word embeddings which is presented in Table 2. We evaluate the performance using **Homogeneity** (each cluster contains only members of a single class) and **Completeness** (all members of a given class are assigned to the same cluster) from (Rosenberg and Hirschberg, 2007) which is presented in Table 1. As seen from Table 1, **fastText** performs very well when the number of clusters are large compare to other embeddings.

4.1.3. Baselines

- *K-means* (Wagstaff et al., 2001) on original keyword features which are weighted with Term Frequency-Inverse Document Frequency (TF-IDF).
- *Spectra*
algorith

[3] https://github.com/jacoxu/StackOverflow
[4] https://www.kaggle.com/c/predict-closed-questions-onstack-overflow/download/train.zip
[5] http://jwebpro.sourceforge.net/data-web-snippets.tar.gz

Table 1
Results of homogeneity and completeness with different pre-trained word embeddings.

Word Embeddings	StackOverflow		SearchSnippets	
	Homogeneity (%)	Completeness (%)	Homogeneity (%)	Completeness (%)
Word2Vec	26.3	26.3	57.7	58.7
GloVe	44.1	47.0	**62.3**	**58.8**
fastText	**66.6**	**70.0**	57.0	57.6

Table 2
Comparisons of ACC and NMI of clustering methods on two public datasets.

Methods	StackOverflow		SearchSnippets	
	ACC (%)	NMI (%)	ACC (%)	NMI (%)
K-means (Wagstaff et al., 2001)	20.31	15.64	33.77	21.40
Spectral Clustering (Belkin and Niyogi, 2001)	27.55	21.03	63.90	48.44
Average Embedding	37.22	38.43	64.63	50.59
STCC (Xu et al., 2015)	51.13	49.03	77.09	63.16
STCC-2 (Xu et al., 2017)	51.20	49.09	77.08	62.99
HierGRU + GloVe	54.5	48.8	81.0	60.3
HierGRU + fastText	**81.2**	**65.0**	**82.5**	**64.7**

- *Average embedding:* We take the pre-trained word embeddings (Bojanowski et al., 2017) of all the non stopwords in a sentence and take the weighted vector average according to the term-frequency (TF) of a word in a sentence then run K-means on it.
- *STCC* (Xu et al., 2015) integrates the ability of convolutional filters to capture local features for high-quality text representation into a self-taught learning framework (Zhang et al., 2010) to cluster short texts.
- *STCC-2* (Xu et al., 2017) incorporates some semantic features and learn non-biased deep text representation in an unsupervised manner using self-taught Convolutional Neural Networks (CNN).

4.1.4. Results

We present our experimental results compared to different simple and state-of-the-art baselines in Table 2. We evaluate clustering performance using the accuracy (**ACC**) and the normalized mutual information metric (**NMI**) (Cai et al., 2005). Here ACC is the ration between $\boldsymbol{c_0}$ and **N**, where $\boldsymbol{c_0}$ is the the number of the text matched between the obtained cluster and the cluster provided by the corpus and **N** is the total number of text. **NMI** is the normalized mutual information between the obtained cluster and the corpus cluster. Normalization is to calculate the Mutual Information (MI) score to scale the results between 0 (no mutual information) and 1 (perfect correlation) between two clusters. According to the Table 2, our model achieves the best clustering performance on all the metrics for both the datasets using **fastText** word embeddings.

4.2. Neural sentence fusion

4.2.1. Dataset

Training set: Neural **seq2seq** encoder-decoder models are usually trained with lots of human-generated references, but there are very few gold references available for the multi-sentence compression task provided by McKeown et al. (2010); Toutanova et al. (2016), which are largely insufficient for training our Neural Sentence Fusion model. Therefore, we use CNN/DailyMail corpus (Hermann et al., 2015) to automatically construct our training set. The CNN/DailyMail dataset contains almost 312 K documents, each with 3-4 highlights that summarize the contents of the article. We take each highlight sentence and map it with the document sentences using word overlap based on Jaccard Similarity. We set a similarity threshold ($t = 0.25$) by using a hold out dataset SICK[6] of SemEval-2014 (Marelli et al., 2014). We take only the many-to-one mappings which involve multiple source sentences from a docu-

Table 3
Statistics of the training set.

Total no of Training Samples (*pairs*)	680,367
Avg. Source Length (*words*)	23.25
Avg. Target Length (*words*)	12.71
Avg. Source to Target Ratio (*sentences*)	3 : 1
Total no of Vocabulary (*words*)	128K

ment and filtered out the rest. Our resulting training set contains 680,367 pairs of multiple source sentence to one target sentence pairs. We present a statistics of our training set in Table 3.

SFC test set: We use the human generated sentence fusion dataset released by McKeown et al. (2010). This dataset consists of 300 English sentence pairs taken from newswire clusters accompanied by human-produced rewrites.

MSR-ATC test set: Toutanova et al. (2016) introduced a manually-created, multi-reference dataset for abstractive sentence and short paragraph compression. It contains approximately 6000 source texts with multiple references accompanied by up to five crowd-sourced rewrites. We filtered out the pairs which contain only single source sentence. We obtained 2,405 multiple source sentence pairs with five human reference variations for our testing.

AAES test set: Ouyang et al. (2017) uses trained annotators to generate abstractive summaries of 476 personal narratives imposing five different rewriting operations reduction (compression), combination (fusion), syntactic transformation, lexical paraphrasing and generalization/ specification to rewrite each extracted phrases. We are not aware of any published results on this recently released dataset.

4.2.2. Baselines

The Word-Graph based approaches that only require a POS tagger was first proposed by (Filippova, 2010). The compressed sentences are generated by finding k-shortest paths in the word graph. (Boudin and Morin, 2013) improved (Filippova, 2010) approach by re-ranking the fusion candidate paths according to keyphrases. However, they reported that the generated sentences were missing important information and were not grammatical. Except (Filippova, 2010; Boudin and Morin, 2013), we didn't find any competitive baseline for this specific task to compare with our model. Furthermore, we implement a sequence-to-sequence model with attention following Luong et al. (2015)'s method as our baseline and denote it as **"s2s+att"**.

4.2.3. Evaluation metric

We evaluate our system automatically using various automatic metrics as described below.

BLEU (Papineni et al., 2002) relies on exact matching of *n*-grams and has no concept of paraphrasing. We used the implementation provided in NLTK[7] considering up to 4-gram matching.

METEOR (Denkowski and Lavie, 2014) where the alignment is based on exact token matching, followed by WordNet synonyms, stemmed tokens and then look-up table paraphrases.

Compression ratio (CR) is a measure of how terse a compression. A compression ratio of zero implies that the source sentence is fully uncompressed.

Copy rate: We define copy rate as how many tokens are copied to the abstract sentence from the source sentence without paraphrasing using the following equation. Lower copy rate score means more paraphrasing is involved in the abstract sentence.

$$Copy\ Rate = \frac{|S_{orig} \cap S_{abs}|}{|S_{abs}|}.$$

Furthermore, we also use Embedding Average Cosine Similarity (**EACS**) and Greedy Matching Score (**GMS**)[8] from Sharma et al. (2017)'s method to measure the abstractiveness of our generated outputs which have stronger correlation with human reference.

[6] http://clic.cimec.unitn.it/composes/sick.html
[7] https://github.com/nltk/nltk/tree/develop/nltk/translate

Table 4
Performance of different systems compare to our proposed Neural Sentence Fusion **(NeuFuse)** model.

Datasets	Models	BLEU	METEOR	CR	Copy Rate	GMS	EACS
SFC	(Filippova, 2010)	42.07	34.10	57.57	99.84	84.30	88.94
	(Boudin and Morin, 2013)	44.64	35.12	37.95	100	80.00	86.79
	(Banerjee et al., 2015)	42.30	34.3	44.9	99.8	77.86	83.92
	(Nayeem and Chali, 2017)	42.5	43.70	41.95	76.2	76.78	81.45
	s2s+att (our baseline)	56.25	37.54	62.31	97.92	85.60	89.35
	NeuFuse	**61.39**	**38.49**	**66.93**	**90.30**	**90.37**	**92.81**
MSR-ATC	(Filippova, 2010)	40.95	35.91	67.04	99.91	85.31	88.47
	(Boudin and Morin, 2013)	43.74	36.62	41.00	100	82.15	90.76
	s2s+att (our baseline)	49.87	37.05	65.63	97.87	87.64	91.25
	NeuFuse	**52.49**	**37.48**	**69.96**	**86.28**	**89.67**	**93.97**
AAES	(Filippova, 2010)	10.97	16.24	**82.91**	99.39	65.39	84.46
	(Boudin and Morin, 2013)	10.67	14.38	80.06	100	67.93	85.46
	s2s+att (our baseline)	26.52	18.08	62.87	98.54	69.90	87.74
	NeuFuse	**28.85**	**19.87**	58.99	**86.48**	**71.46**	**90.05**

4.2.4. Results

We report our system (**NeuFuse**) performance compared with the baselines in Table 4. Our model jointly improves the information coverage (BLEU, GMS) and complete abstractiveness (METEOR, Copy Rate, EACS) with a balanced compression ratio (CR). Instead of over compressing the generated sentences our model try to balance the information coverage with CR for long input sentences such as narratives from **AAES** test set. Copy Rate scores of other baseline systems clearly indicate the fact that they are doing completely deletion based compression, no new words or words with morphological variation are generated in the process. We present some randomly selected outputs generated by our model for both the datasets in Table 5.

4.2.5. Human evaluation

For Neural Sentence Fusion evaluation we have randomly selected 10 documents[9] and gave it to 27 people from all over the world. They have evaluated each document in three different aspects i.e., Content, Readability and Overall. They have evaluated each summary with score for each aspect from 1 to 5, where 1 is represented as very poor performance and 5 as very good performance. Here **content** means how well the summary can represent the meaning of the main document, **readability** represents the grammatical perfection and the sentence structure of the summary and **overall** is to evaluate the summary based on the previous two. We collected their individual evaluation and calculate the average rating given by all of them. The results are shown in Table 6. None of the authors have participated in this evaluation.

4.3. Multi-document summarization

4.3.1. Dataset

We consider the generic multi-document summarization dataset provided at Document Understanding Conference (DUC 2004) containing 50 document clusters. The Opinosis (Ganesan et al., 2010) is another dataset consists of short user reviews in 51 different topics collected from TripAdvisor, Amazon, and Edmunds.

4.3.2. Evaluation metric

We evaluate our summarization system using **ROUGE**[10] (Lin, 2004) on DUC 2004 (Task-2, Length limit (L) = 100 Words) and Opinosis 1.0 (L = 15 Words). We report limited length recall performance for both the metrics, as our system generated summaries are forced to be concise through some constraints (such as length limit constraint). Our results include R-1, R-2, and R-SU4, which counts matches in unigrams, bigrams, and skip-bigrams, respec-

[8] https://github.com/Maluuba/nlg-eval
[9] https://github.com/Anonymous3058/Elsvier-Human-Evaluation

Table 5
Randomly selected outputs for our **NeuFuse** model form different test sets. Green Shading intensity represents new word generation other than source input sentence words and Yellow Shading intensity represents the morphological variation generation from the source input sentence words.

	MSR-ATC
Input Sentences	Will the administration live up to its environmental promises ? Can we save the last of our ancient forests from the chainsaw ?
Reference (*best*)	Will the administration live up to its environmental promises to save our ancient forests?
System Output	Officials could save the last of our ancient forests from the chainsaw.
	SFC
Input Sentences	Senators and Obama had stood by him, but Daschle withdrew today, saying he did not want to be a distraction. Asked about the stunning reversal, White House spokesman Robert Gibbs said Daschle made the decision because he did not want to be a distraction to Obama's agenda.
Reference (*best*)	Daschle made the decision because he did not want to be a distraction.
System Output	Daschle said he did not want to be a distraction in Obama's agenda.
	AAES
Input Sentences	My girlfriend told me the week before I went to college that I got her pregnant knowing that we were not going to date long distance. She sent me a picture of a stock photo pregnancy test she cropped that was positive I literally just Google searched "Positive Pregnancy Tests" and it was one of the first ones. I made her come over and take another one. It was negative, she left.
Reference (*best*)	My girlfriend lied about having a positive pregnancy test using an image she found on Google.
System Output	My girlfriend literally Google searched "Positive Pregnancy Tests Images".

Table 6
Human evaluation summary for single document summaries.

Datasets	Content	Readability	Overall
MSR-ATC	3.63	3.78	3.69
SFC	3.70	3.73	3.74
AAES	3.53	3.88	3.73
Average Performance	3.62	3.80	3.72

tively.

4.3.3. Baseline systems

The summaries generated by the **LexRank** (Erkan and Radev, 2004) and the state-of-the-art summarizers (**Submodular** (Lin and Bilmes, 2011) and **RegSum** (Hong and Nenkova, 2014)) on the DUC 2004 dataset were collected from Hong et al. (2014). In the case of **ILPSumm** (Banerjee et al., 2015) and **PDG*** (Yasunaga et al., 2017), we use the author provided implementation to generate summary from their model. We use the paper reported scores for **NAMDS** (Zhang et al., 2018). For Opinosis 1.0 dataset, we use an open source implementation of **TextRank** (Mihalcea and Tarau, 2004). Moreover, we use the author provided implementation for the **Opinosis** (Ganesan et al., 2010) and **Biclique** (Muhammad et al., 2016) to generate summaries.

4.3.4. Results

According to the Table 7 & 8, our multi-document level model achieves the best summarization performance on all the ROUGE metrics for both the datasets. However, ROUGE scores are unfairly biased towards lexical overlap at surface level. Therefore, unable to measure the abstractiveness property. Taking this into account, we use document level **EACS** (Sharma et al., 2017) which considers word embeddings to compute the semantic similarity of the words. Moreover, we verify the copy rate scores of the human summary and our system generated summary with the source documents. According to Table 9, our system generated summary is very close to human references in terms of both **EACS** and **Copy Rate** scores.

4.3.5. Human evaluation

For Multi-Document Summary evaluation we have randomly selected 6 documents[11] and gave it to 3 people who has a Masters degree in Computer Science from North America. As our main focus is on the MDS task, we tried to select people who has experience of this kind of scenarios. We collected their individual evaluation and calculate

[10] ROUGE-1.5.5 with options: -n 2 -m -u -c 95 -x -r 1000 -f A -p 0.5 -t 0.

Table 7
Comparison results on the **DUC 2004** test set.

DUC 2004			
Models	R-1	R-2	R-SU4
LexRank (Erkan and Radev, 2004)	35.95	7.47	12.48
Submodular (Lin and Bilmes, 2011)	39.18	9.35	14.22
RegSum (Hong and Nenkova, 2014)	38.57	9.75	13.81
ILPSumm (Banerjee et al., 2015)	39.24	11.99	14.76
PDG* (Yasunaga et al., 2017)	38.45	9.48	13.72
NAMDS (Zhang et al., 2018)	36.70	7.83	12.40
ParaFuse_Doc (Nayeem et al., 2018)	40.07	12.04	14.24
NeuFuse_multidoc (ours)	**41.92**	**12.22**	**15.59**

Table 8
Comparison results on the **Opinosis 1.0**.

Opinosis 1.0			
Models	R-1	R-2	R-SU4
TextRank (Mihalcea and Tarau, 2004)	27.56	6.12	10.53
Opinosis (Ganesan et al., 2010)	32.35	9.13	14.35
Biclique (Muhammad et al., 2016)	33.03	8.96	14.18
ParaFuse_Doc (Nayeem et al., 2018)	33.86	9.74	xxy
NeuFuse_multidoc (ours)	**43.98**	**17.31**	**22.19**

Table 9
Abstractiveness property.

Dataset	Copy rate		EACS
	Human Summary	System Summary	
DUC 2004	76.22	88.01	95.46
Opinosis 1.0	69.58	70.48	88.28

Table 10
Human evaluation summary for multi-document summaries.

Datasets	Content	Readability	Overall
DUC2004	4.00	4.44	3.89
Opinosis	3.89	4.44	3.67
Average Performance	3.95	4.44	3.78

the average rating given by all of them. The results are shown in Table 10. They have evaluated each document in three different aspects i.e., Content, Readability and Overall. They have evaluated each summary with score for each aspect from 1 to 5, where 1 is represented as very poor performance and 5 as very good performance. Here **content** means how well the summary can represent the meaning of the main document, **readability** represents the grammatical perfection and the sentence structure of the summary and **overall** is to evaluate the summary based on the previous two. None of the authors has participated in this evaluation.

5. Conclusion & future work

In this paper, our contributions were, (a) an unsupervised, simple sentence clustering model which outperform several popular clustering methods; (b) our neural sentence fusion model which was the first to investigate adapting

[11] https://github.com/Anonymous3058/Elsvier-Human-Evaluation

neural models to sentence fusion task and (c) our main model which integrates three important measures namely importance, coverage, and diversity under a desired length limit. Our system has achieved the state-of-the-art results while tested on two different datasets. We adapted our proposed unsupervised sentence clustering model and neural sentence fusion model to the task of abstractive multi-document summarization. In future, we will broadly focus on contributing a better neural architecture to encode a multi-document set. Furthermore, we will try to propose a new sentence abstraction technique (e.g., syntactic reorganization) using bi-directional beam search.

Acknowledgment

The research reported in this paper was conducted at the University of Lethbridge and supported by the Natural Sciences and Engineering Research Council (NSERC) of Canada discovery grant and the University of Lethbridge.

References

Aggarwal, C.C., Zhai, C., 2012. A survey of text clustering algorithms. In: Aggarwal, C.C., Zhai, C. (Eds.), Springer US, Boston, MA, pp. 77–128.

Bahdanau, D., Cho, K., Bengio, Y., 2015. Neural machine translation by jointly learning to align and translate. In: Proceedings of the ICLR 2015.

Banerjee, S., Mitra, P., Sugiyama, K., 2015. Multi-document abstractive summarization using ILP based multi-sentence compression. In: Proceedings of the 24th International Conference on Artificial Intelligence. AAAI Press, pp. 1208–1214.

Barzilay, R., McKeown, K.R., 2005. Sentence fusion for multidocument news summarization. Comput. Linguist. 31 (3), 297–328. doi: 10.1162/089120105774321091.

Belkin, M., Niyogi, P., 2001. Laplacian eigenmaps and spectral techniques for embedding and clustering. In: Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic. MIT Press, Cambridge, MA, USA, pp. 585–591.

Bojanowski, P., Grave, E., Joulin, A., Mikolov, T., 2017. Enriching word vectors with subword information. Trans. Assoc. Comput. Linguist. 5, 135–146.

Boudin, F., Morin, E., 2013. Keyphrase extraction for n-best reranking in multi-sentence compression. In: Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Association for Computational Linguistics, Atlanta, Georgia, pp. 298–305.

Cai, D., He, X., Han, J., 2005. Document clustering using locality preserving indexing. IEEE Trans. Knowl. Data Eng. 17 (12), 1624–1637. doi: 10.1109/TKDE.2005.198.

Cao, Z., Wei, F., Li, W., Li, S., 2018. Faithful to the original: fact aware neural abstractive summarization. In: Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, New Orleans, Louisiana, USA, February 2-7, 2018.

Celikyilmaz, A., Bosselut, A., He, X., Choi, Y., 2018. Deep communicating agents for abstractive summarization. In: Proceedings of the NAACL 2018 - Conference of the North American Chapter of the Association for Computational Linguistics.

Cho, K., van Merrienboer, B., Bahdanau, D., Bengio, Y., 2014. On the properties of neural machine translation: encoder−decoder approaches. In: Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation. Association for Computational Linguistics, pp. 103–111. doi: 10.3115/v1/W14-4012.

Chopra, S., Auli, M., Rush, A.M., 2016. Abstractive sentence summarization with attentive recurrent neural networks. In: Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Association for Computational Linguistics, San Diego, California, pp. 93–98.

Denkowski, M., Lavie, A., 2014. Meteor universal: language specific translation evaluation for any target language. In: Proceedings of the Ninth Workshop on Statistical Machine Translation. Association for Computational Linguistics, Baltimore, Maryland, USA, pp. 376–380.

Erkan, G., Radev, D.R., 2004. Lexrank: graph-based lexical centrality as salience in text summarization. J. Artif. Int. Res. 22 (1), 457–479.

Fan, F., Grangier, D., Auil, M., 2018. Controllable abstractive summarization. Proceedings of the 2nd Workshop on Neural Machine Translation and Generation 45–54.

Filippova, K., 2010. Multi-sentence compression: finding shortest paths in word graphs. In: Proceedings of the 23rd International Conference on Computational Linguistics. Association for Computational Linguistics, Stroudsburg, PA, USA, pp. 322–330.

Ganesan, K., Zhai, C., Han, J., 2010. Opinosis: a graph-based approach to abstractive summarization of highly redundant opinions. In: Proceedings of the 23rd International Conference on Computational Linguistics. Association for Computational Linguistics, Stroudsburg, PA, USA, pp. 340–348.

Gillick, D., Favre, B., 2009. A scalable global model for summarization. In: Proceedings of the Workshop on Integer Linear Programming for Natural Langauge Processing. Association for Computational Linguistics, Stroudsburg, PA, USA, pp. 10–18.

Hermann, K.M., Kočiský, T., Grefenstette, E., Espeholt, L., Kay, W., Suleyman, M., Blunsom, P., 2015. Teaching machines to read and comprehend. In: Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1. MIT Press, Cambridge, MA, USA, pp. 1693–1701.

Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. Neural Comput. 9 (8), 1735–1780.

Hong, K., Conroy, J., Favre, B., Kulesza, A., Lin, H., Nenkova, A., 2014. A repository of state of the art and competitive baseline summaries for generic news summarization. In: Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14). European Language Resources Association (ELRA), Reykjavik, Iceland, pp. 1608–1616.ACL Anthology Identifier: L14-1070.

Hong, K., Nenkova, A., 2014. Improving the estimation of word importance for news multi-document summarization. In: Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics. Association for Computational Linguistics, Gothenburg, Sweden, pp. 712–721.

Li, P., Lam, W., Bing, L., Guo, W., Li, H., 2017. Cascaded attention based unsupervised information distillation for compressive summarization. In: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, Copenhagen, Denmark, pp. 2081–2090.

Li, P., Lam, W., Bing, L., Wang, Z., 2017. Deep recurrent generative decoder for abstractive text summarization. In: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, Copenhagen, Denmark, pp. 2091–2100.

Lin, C.-Y., 2004. Rouge: a package for automatic evaluation of summaries. In: Marie-Francine Moens, S.S. (Ed.), Text Summarization Branches Out: Proceedings of the ACL-04 Workshop. Association for Computational Linguistics, Barcelona, Spain, pp. 74–81.

Lin, H., Bilmes, J., 2011. A class of submodular functions for document summarization. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1. Association for Computational Linguistics, Stroudsburg, PA, USA, pp. 510–520.

Luong, M.-T., Pham, H., Manning, C.D., 2015. Effective approaches to attention-based neural machine translation. Empirical Methods in Natural Language Processing (EMNLP). Association for Computational Linguistics, Lisbon, Portugal, pp. 1412–1421.

Ma, S., Sun, X., Xu, J., Wang, H., Li, W., Su, Q., 2017. Improving semantic relevance for sequence-to-sequence learning of chinese social media text summarization. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). Association for Computational Linguistics, Vancouver, Canada, pp. 635–640.

Marelli, M., Menini, S., Baroni, M., Bentivogli, L., Bernardi, R., Zamparelli, R., 2014. A sick cure for the evaluation of compositional distributional semantic models. In: Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14). European Language Resources Association (ELRA), Reykjavik, Iceland.

McKeown, K., Rosenthal, S., Thadani, K., Moore, C., 2010. Time-efficient creation of an accurate sentence fusion corpus. Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics. Association for Computational Linguistics, Los Angeles, California, pp. 317–320.

Mihalcea, R., Tarau, P., 2004. Textrank: bringing order into texts. In: Lin, D., Wu, D. (Eds.), Proceedings of EMNLP 2004. Association for Computational Linguistics, Barcelona, Spain, pp. 404–411.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G., Dean, J., 2013. Distributed representations of words and phrases and their compositionality. In: Proceedings of the 26th International Conference on Neural Information Processing Systems. Curran Associates Inc., USA, pp. 3111–3119.

Muhammad, A.S., Damaschke, P., Mogren, O., 2016. Summarizing online user reviews using bicliques. In: Proceedings of the 42Nd International Conference on SOFSEM 2016: Theory and Practice of Computer Science - Volume 9587. Springer-Verlag New York, Inc., New York, NY, USA, pp. 569–579. doi: 10.1007/978-3-662-49192-8_46.

Murtagh, F., Legendre, P., 2014. Ward's hierarchical agglomerative clustering method: which algorithms implement ward's criterion? J. Classif. 31 (3), 274–295. doi: 10.1007/s00357-014-9161-z.

Nallapati, R., Zhou, B., dos Santos, C., glar Gulçehre, Ç., Xiang, B., 2016. Abstractive text summarization using sequence-to-sequence RNNS and beyond. CoNLL 2016 280.

Napoles, C., Gormley, M., Van Durme, B., 2012. Annotated gigaword. In: Proceedings of the Joint Workshop on Automatic Knowledge Base Construction and Web-scale Knowledge Extraction. Association for Computational Linguistics, Stroudsburg, PA, USA, pp. 95–100.

Narayan, S., Cohen, S.B., Lapata, M., 2018. Ranking sentences for extractive summarization with reinforcement learning. In: Proceedings of the NAACL 2018 - Conference of the North American Chapter of the Association for Computational Linguistics.

Narayan, S., Papasarantopoulos, N., Cohen, S.B., Lapata, M., 2018. Neural extractive summarization with side information. In: Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence.

Nayeem, M.T., Chali, Y., 2017. Paraphrastic fusion for abstractive multi-sentence compression generation. In: Proceedings of the 2017 ACM on Conference on Information and Knowledge Management. ACM, pp. 2223–2226.

Nayeem, M.T., Fuad, T.A., Chali, Y., 2018. Abstractive unsupervised multi-document summarization using paraphrastic sentence fusion. In: Proceedings of the 27th International Conference on Computational Linguistics, pp. 1191–1204.

Ouyang, J., Chang, S., McKeown, K., 2017. Crowd-sourced iterative annotation for narrative summarization corpora. In: Proceedings of the EACL 2017, Short Papers.

Papineni, K., Roukos, S., Ward, T., Zhu, W.-J., 2002. BLEU: a method for automatic evaluation of machine translation. In: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics. Association for Computational Linguistics, Stroudsburg, PA, USA, pp. 311–318. doi: 10.3115/1073083.1073135.

Paulus, R., Xiong, C., Socher, R., 2018. A deep reinforced model for abstractive summarization. International Conference on Learning Representations https://openreview.net/forum?id=HkAClQgA-.

Pennington, J., Socher, R., Manning, C., 2014. Glove: global vectors for word representation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). Association for Computational Linguistics, Doha, Qatar, pp. 1532–1543.

Phan, X.-H., Nguyen, L.-M., Horiguchi, S., 2008. Learning to classify short and sparse text & web with hidden topics from large-scale data collections. In: Proceedings of the 17th International Conference on World Wide Web. ACM, New York, NY, USA, pp. 91–100. doi: 10.1145/1367497.1367510.

Rosenberg, A., Hirschberg, J., 2007. V-measure: a conditional entropy-based external cluster evaluation measure. In: Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL).

Rush, A.M., Chopra, S., Weston, J., 2015. A neural attention model for abstractive sentence summarization. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, Lisbon, Portugal, pp. 379–389.

See, A., Liu, P.J., Manning, C.D., 2017. Get to the point: summarization with pointer-generator networks. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Association for Computational Linguistics, Vancouver, Canada, pp. 1073–1083.

Sharma, S., Asri, L.E., Schulz, H., Zumer, J., 2018. Relevance of unsupervised metrics in task-oriented dialogue for evaluating natural language generation. https://openreview.net/forum?id=r17lFgZ0Z.

Suzuki, J., Nagata, M., 2017. Cutting-off redundant repeating generations for neural abstractive summarization. In: Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers. Association for Computational Linguistics, Valencia, Spain, pp. 291–297.

Toutanova, K., Brockett, C., Tran, K.M., Amershi, S., 2016. A dataset and evaluation metrics for abstractive compression of sentences and short paragraphs. In: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, Austin, Texas, pp. 340–350.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L.u., Polosukhin, I., 2017. Attention is all you need. In: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (Eds.), Advances in Neural Information Processing Systems 30. Curran Associates, Inc., pp. 5998–6008.

Wagstaff, K., Cardie, C., Rogers, S., Schrödl, S., 2001. Constrained k-means clustering with background knowledge. In: Proceedings of the Eighteenth International Conference on Machine Learning. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, pp. 577–584.

Xu, J., Wang, P., Tian, G., Xu, B., Zhao, J., Wang, F., Hao, H., 2015. Short text clustering via convolutional neural networks. In: Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing. Association for Computational Linguistics, pp. 62–69. doi: 10.3115/v1/W15-1509.

Xu, J., Xu, B., Wang, P., Zheng, S., Tian, G., Zhao, J., Xu, B., 2017. Self-taught convolutional neural networks for short text clustering. Neural Netw. 88, 22–31. doi: 10.1016/j.neunet.2016.12.008.

Yasunaga, M., Zhang, R., Meelu, K., Pareek, A., Srinivasan, K., Radev, D., 2017. Graph-based neural multi-document summarization. In: Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017). Association for Computational Linguistics, Vancouver, Canada, pp. 452–462.

Zhang, D., Wang, J., Cai, D., Lu, J., 2010. Self-taught hashing for fast similarity search. In: Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM, New York, NY, USA, pp. 18–25. doi: 10.1145/1835449.1835455.

Zhang, J., Tan, J., Wan, X., 2018. Towards a neural network approach to abstractive multi-document summarization. arXiv:1804.09010.

Zhou, Q., Yang, N., Wei, F., Zhou, M., 2017. Selective encoding for abstractive sentence summarization. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Association for Computational Linguistics, Vancouver, Canada, pp. 1095–1104.